



VAASAN AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES

Leeni Liimakka

# JÄRJESTELMÄN SUUNNITTELU

Tietojenkäsittely  
2018

## TIIVISTELMÄ

Tekijä	Leeni Liimakka
Opinnäytetyön nimi	Järjestelmän suunnittelu
Vuosi	2018
Kieli	suomi
Sivumäärä	38 + 1 liite
Ohjaaja	Sirkka Hellman

---

Digitaalisuutta voidaan käyttää hyväksi tuomaan ihmisiä lähemmäksi luontoa. Tämän opinnäytetyön tavoitteena oli tehdä suunnitelmat sovellukselle tai järjestelmälle, joka näyttää käyttäjälle lähimpien laavupaikkojen sijainnit kartalla.

Kyseinen sovellus tulisi olemaan ainoa ilmainen laavujen hakusovellus Suomessa. Tavoitteena on tehdä siitä myös käyttäjäystävällisemmän, mitä saatavilla olevat sovellukset ovat. Nyt olemassa olevat sivustot eivät joko ole mobiilikäyttöön soveltuvia tai sitten ne ovat maksullisia.

Tämä suunnitelma on ensimmäinen vaihe sovelluksen toteuttamiselle. Tästä voidaan jatkaa ohjelmoinnilla eteenpäin. Tähän työhön on tehty hahmotelmat käyttöliittymästä, miltä sivustot näyttävät, millainen tietokannan tulisi olla sekä käyttäjätarinat.

## ABSTRACT

Author	Leeni Liimakka
Title	System design
Year	2018
Language	Finnish
Pages	38 + 1 Appendix
Name of Supervisor	Sirkka Hellman

---

Digitalization can be used to bring people closer to nature. The aim for this thesis was to design and make plans for an app or a website that shows all the lean-tos close to the user on a map.

This app would be the only free and user friendly search tool for lean-to locations in Finland. The current lean-to apps are either not suitable for mobile usage or a user has to pay to use them.

The system design is the first phase for the development of the app. The programming itself can be carried out after this. The included are drafts for user interface designs and wireframes, the database behind the app and user stories.

---

Keywords	Project plan document, software development, customer requirements, functional specifications
----------	--

# SISÄLLYS

## TIIVISTELMÄ

## ABSTRACT

1	JOHDANTO .....	8
1.1	Työn tausta ja aihe .....	8
1.2	Työn tavoite .....	9
1.3	Työn rakenne .....	10
2	JÄRJESTELMÄN TARKOITUS .....	12
2.1	Järjestelmän yleiskuvaus ja tarvekartoitus .....	12
2.2	Laite- ja ohjelmistoympäristö .....	13
3	VAATIMUSMÄÄRITTELYT .....	15
3.1	Käyttäjäkertomukset .....	15
3.1.1	Scrum .....	15
3.1.2	Käyttäjäkertomus – user story .....	16
3.2	UML-notaatiostandardi .....	19
3.3	Käyttötapaus .....	19
3.3.1	Käyttötapauskaaviot - use case diagram .....	19
3.3.2	Käyttäjien analysointi ja käyttötapaukset – use cases .....	21
4	TIETOKANTA .....	26
4.1	Taulut .....	26
4.2	Tietotyypit .....	29
5	VISUAALINEN SUUNNITTELU .....	32
5.1	Wireframe – rautalankamalli .....	32
5.2	Fontti ja värimaailma .....	33
5.3	Visuaalinen suunnittelu .....	34
5.3.1	Invision .....	35
6	POHDINTA .....	37
	LÄHTEET .....	38

## LIIETTEET

## KUVIO- JA TAULUKKOLUETTELO

<b>Kuvio 1.</b> Perinteinen laavu.	9
<b>Kuvio 2.</b> Ohjelmistotuotannon osa-alueet (Haikala & Märijärvi 2002, 35).	12
<b>Kuvio 3.</b> Suosituimmat mobiililaitteiden käyttöjärjestelmät (IDC 2017).	14
<b>Kuvio 4.</b> Scrum-malli (Kasurinen 2013).	16
<b>Kuvio 5.</b> Esimerkki käyttötapauskaaviosta (Xiacong 2015, 204).	20
<b>Kuvio 6.</b> Laavusovelluksen käyttötapauskaavio.	21
<b>Kuvio 7.</b> Laavusivuston hahmotelma, jossa kaksi päätaulua.	27
<b>Kuvio 8.</b> Laavusivuston tietokannan luokkakaavio.	28
<b>Kuvio 9.</b> Muokattu luokkakaavio laavusivuston tietokannasta.	29
<b>Kuvio 10.</b> Laavutaulun tiedot sekä pääavain.	30
<b>Kuvio 11.</b> Tietokannan taulukaavio DdDesignerissa.	31
<b>Kuvio 12.</b> Karkea ensimmäinen rautalankamallisuunnitelma sivustosta.	33
<b>Kuvio 13.</b> Värien suunnittelua Adobe Color CC -ohjelmassa.	34
<b>Kuvio 14.</b> Laavusovelluksen kaksi eri logo-vaihtoehtoa.	34
<b>Kuvio 15.</b> Laavusovelluksen oletusnäkökulma puhelimessa.	35
<b>Kuvio 16.</b> Invisionin näkökulma.	36
<b>Taulukko 1.</b> Laavusivustoon tekemäni käyttäjäkertomukset ja -tarinat.	17
<b>Taulukko 2.</b> Listanäkymä laavun tiedoista.	27
<b>Taulukko 3.</b> Luokkakaavion merkintöjen selitykset.	28
<b>Taulukko 4.</b> Tietotyyppejä selityksineen.	29

## **LIITELUETTELO**

**LIITE 1.** Photoshopilla tehdyt laavusovelluksen eri näyttöruudut.

# 1 JOHDANTO

Älypuhelimet, uusi teknologia ja digitaalisuus ovat muokanneet maailmaa pienemmäksi. Toisella puolella maapalloa olevien ystävien kanssa kommunikointi on helppoa ja vaivatonta. Nämä laitteet sekä sosiaalinen media ovat muuttaneet tapaamme kuluttaa vapaa-aikaamme. Kotona oleminen on aivan liian houkuttelevaa. Miksi lähteä ulos ja luontoon, kun voimme sormen painalluksella tehdä niin paljon enemmän kotisohvalla? Onneksi digitaalisuutta voidaan käyttää myös ihmisten liikkumisen ja ulkoilemisen hyväksi.

Sovelluskaupoista löytyy erilaisia vaihtoehtoja luonnossa liikkumiseen. On karttoja, kasvihakuja, luonnonläheisiä kaupunkikierroksia, metsäreittejä lastenkirjojen hahmojen löytämiseksi sekä tähtikarttoja. Nämä kaikki yrittävät teknologian avulla tuoda luontoa lähemmäksi ihmistä.

Jokamiehenoikeus on harvinaislaatuinen etuoikeus, jollaista ei monessa muussa maassa ole. Suomen kaunis luonto antaa hyvät puitteet siellä liikkumiseen ja oleskelemiseen. Kaikilla Suomessa oleskelevilla ihmisillä on sekä oikeus että velvollisuus käyttää jokamiehenoikeuksia. Liikkuminen, marjastaminen, sienestäminen ja telttailu ovat kaikki osa näitä oikeuksia. Sen sijaan roskaaminen häiriön aiheuttaminen, sammaleen tai jäkälän kerääminen tai avotulen tekeminen luvatta ovat kiellettyjä jokamiehenoikeuksissa. Usein retkeilyalueilla on tehty erillisiä tulentekopaikkoja ja monesti näissä paikoissa on kiinteitä laavuja.

## 1.1 Työn tausta ja aihe

Laavut on tehty kaikille retkeilijöille, ulkoilijoille tai kalastajille, jotka tarvitsevat turvallisen paikan nukkua tai levähtää (Kuvio 1). Tulentekopaikalla voi grillata tai keittää kahvia. Usein on joku, joka ylläpitää näitä laavuja, tuo muun muassa polttopuut niihin. Joskus tämä taho on kyläseura tai yksityinen ihminen.

Laavujen sijainnit ovat isommissa retkeilyalueilla tai luontopoluilla merkitty karttoihin ja verkkosivuille, mutta kaikkien laavujen sijainnit eivät ole löydettävissä



kartoista tai sivustoilta lainkaan. Ulkoilijoita ja retkeilijöitä auttaisi, jos nämä löytyisivät yhdestä paikasta näppärästi karttaa selailemalla.



**Kuvio 1.** Perinteinen laavu.

Opinnäytetyöni tavoite on suunnitella laavusovellus tai -sivusto kaikkia ulkoilijoita ja retkeilijöitä varten. Useimmiten laavujen sijainnit ovat löydettävissä suustasuuhun-periaatteella. Tämän sivuston tarkoitus on saada sijainnit näistä yhtenäiseen karttaan kaikkine tarpeellisine tietoineen. Retkeilijät ja ulkoilijat voivat lisätä ja hakea laavujen sijainteja kartalta sekä lukea tietoja niistä.

Kyseisen sovelluksen toteuttaminen on ollut suunnitelmissani jo aiemmin. Opinnäytetyöni aiheeksi tämä oli siten luonnollinen valinta. Tässä työssä työn teettäjää ei varsinaisesti ole, vaan se tulee itselleni myöhempää tarkoitusta varten.

## **1.2 Työn tavoite**

Työn tavoite on tarpeeksi kattavan suunnitelman tekeminen. Opinnäytetyössä tulee olemaan monta eri osa-aluetta, joista keskeisimpinä haasteina tai kysymyksinä pitäisin näitä:

- Miten suunnitella toimiva, responsiivinen sivusto?

- Millainen sovelluksen tietokannan tulee olla toimiakseen hyvin?
- Mitä vaatimusmäärittelyjä sovelluksella on?

Sovelluksen tulee olla responsiivinen ja siten sen tulee toimia varsinkin puhelimen näytöllä vaivattomasti ja sujuvasti. Tietokantasuunnitelmaa varten on käytävä läpi kaikki käyttötapaukset ja käyttäjäkertomukset, jolloin tiedetään, millaisia tietoja tarvitsemme tässä sovelluksessa.

Tiedot sekä taidot opinnäytetyön suunnitteluun tulevat enimmäkseen opinnoistani, jossa suoritin erilaisia verkkopalveluiden ja tietokantojen suunnittelukursseja. Lähteinä käytin sekä verkkosivuja että kirjoja, joista osa oli elektronisia kirjoja. Tietokannan suunnitteluun löysin oppaan: Tietokantojen suunnittelu & indeksointi. Käyttötapauksiin käytin muun muassa Use Cases: Requirements in Context -kirjaa. Yksi mielenkiintoisimmista kirjoista oli Chelsea Myersin kirjoittama Responsive Web Design Patterns.

Sovelluksen toteutus tulee olemaan aikaa vievää ja siksi sen tulee olla hyvin suunniteltu. Työssäni teenkin yksityiskohtaiset suunnitelmat ja ohjeet siten, ettei toteutuksen aloittaminen ole kynnyskysymys ja tiedetään, mikä lopullisen sovelluksen tavoite on.

Laavuista on olemassa sivustoja, mutta panostaisin tässä sovelluksessa karttahakuun enemmän, lisätietoihin laavujen kohdalla ja yleiseen käytettävyyteen. Olen myös sitä mieltä, että sivuston käytön tulee olla ilmaista.

### **1.3 Työn rakenne**

Raportti sisältää yleiskuvauksen sovelluksesta tai sivustosta luvussa kaksi. Tarvekartoituksessa kerrotaan, millaisia vaatimuksia käyttäjillä tulee olemaan. Olen läpikäynyt myös laite- ja ohjelmistoympäristövaatimuksia.

Luvun kolme vaatimusmäärittelyssä käydään läpi tarkemmin eri tarinoita käyttäjän näkökulmasta, millaisia tapauksia tämän sovelluksen käytössä voisi olla. Luettelen useita erilaisia käyttäjäkertomuksia ja käyttötapauksia kaavioineen.

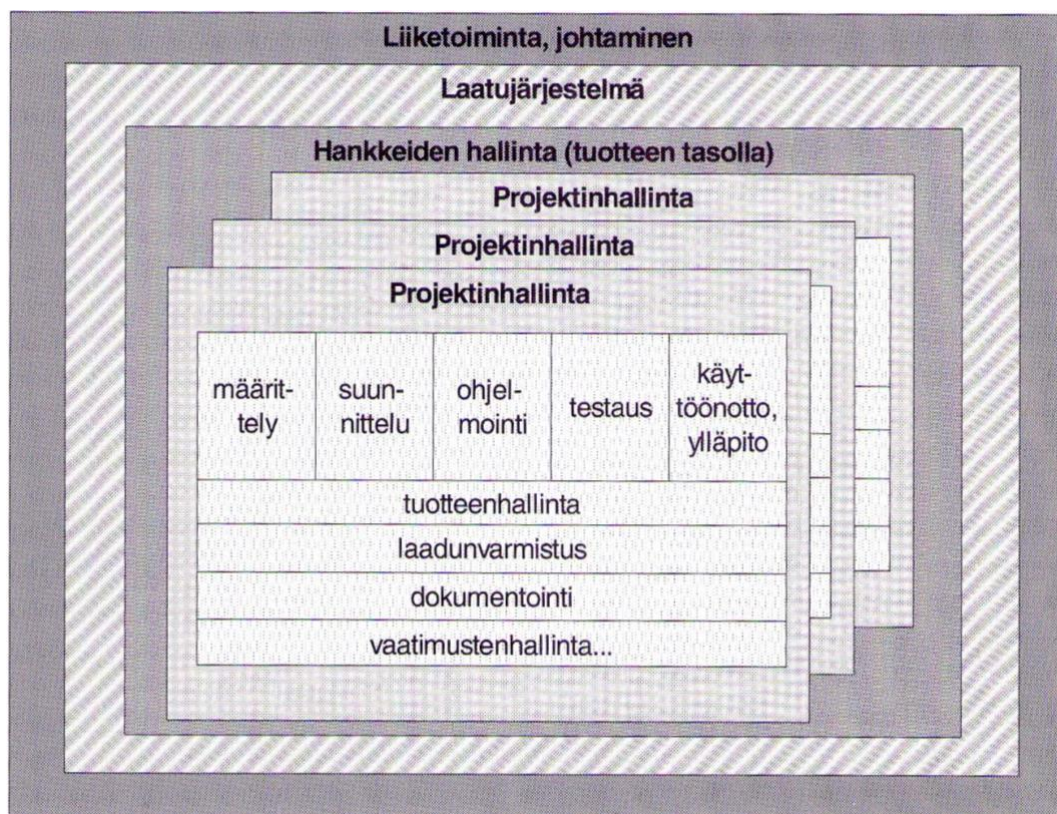
Tietokantaosiossa luvussa neljä tärkeintä on eri taulujen luominen. Käyttötapauskaaviota ja käyttäjätarinoita pohjana käyttäen tietokannan suunnitteleminen sujuu parhaiten. Taulujen hahmottamista auttaa myös, jos se tehdään graafisesti.

Myös visuaalinen suunnittelu tulee olemaan osa tätä työtä. Luvussa viisi suunnitellaan ja esitellään sekä värit, logo että ulkonäkö ylipäätään. Työn loppuosassa onkin näkymät näytöistä rautalankamalleineen.

## 2 JÄRJESTELMÄN TARKOITUS

Laavusovellus tai laavusivustot ovat tarkoitettu kaikille, jotka ovat kiinnostuneita luonnosta, ulkoilusta ja retkeilystä ja haluavat löytää uusia laavupaikkoja, missä voi pitää tauon sekä grillata ja jopa yöpyä. He haluavat jakaa tietojaan muista laavupaikoista muille samankaltaisille ihmisille.

Tässä työssä käyn läpi järjestelmän määrittelyä ja suunnittelua. Ne ovat osa ohjelmistotuotantoa, jossa muita osa-alueita ovat ohjelmointi, testaus sekä käyttöönotto ja ylläpito (Kuvio 2). Järjestelmän määrittelyssä vastataan kysymykseen *mitä* ja suunnittelussa, *miten* (Haikala & Märijärvi 2002, 62).



**Kuvio 2.** Ohjelmistotuotannon osa-alueet (Haikala & Märijärvi 2002, 35).

### 2.1 Järjestelmän yleiskuvaus ja tarvekartoitus

Laavusivuston avulla voidaan selata karttaa ja hakea siitä laavujen sijainteja. Kuka tahansa pystyy tekemään tämän kirjautumatta sivuille. Käyttäjillä on mahdollisuus

rekisteröityä ja kirjautua sovellukseen, jolloin he pystyvät lisäämään uusia laavuja sivustolle ja muokkaamaan antamiaan tietojaan.

Sovelluksen on pystyttävä vastaanottamaan tietoa sivuilleen, antamaan kirjautumisoikeudet käyttäjilleen sekä antamaan lukuoikeudet siihen. Vaatimuksia käyttäjien kannalta on:

- Kirjautuminen sivuille
- Laavujen tietojen haku ja selaus kartalla, myös ilman kirjautumista
- Laavujen lisäys
- Laavujen lisätietojen sekä valokuvien lisäys laavun profiliin
- Lisätyn laavun tietojen muokkaus sekä poistaminen
- Oman profiilitietojen muokkaus sekä poistaminen
- Suosikkilistan tallentaminen
- Unohtuneen salasanan palauttaminen
- Laavujen ja käyttäjien tietojen lisääminen, poistaminen sekä muokkaaminen ylläpitäjän toimesta

Laavusovellus helpottaa ulkoilijoita ja retkeilijöitä löytämään uusia laavupaikkoja haluamaltaan alueelta kartan avulla. Laavut löytyvät kaikki samalta sivustolta ilmaiseksi.

## **2.2 Laite- ja ohjelmistoympäristö**

Pääsääntöisesti laavusivustoa tullaan käyttämään eri selaimilla mobiililaitteissa. Sivuston toki tulee olla responsiivinen toimiakseen myös muilla laitteilla, kuten kotitietokoneella, hyvin. Järjestelmässä tulee olemaan SQL-tietokanta.

Erillisen sovelluksen tulisi olla toimiva sekä Applen (iOS) että Androidin käyttöjärjestelmissä. Nämä kaksi käyttöjärjestelmää ovat suosituimpia mobiililaitteissa. Kuviossa 3 esitetään, mitkä käyttöjärjestelmät ovat olleet suosituimmat toukokuussa 2017.

Period	Android	iOS	Windows Phone	Others
2016Q1	83.4%	15.4%	0.8%	0.4%
2016Q2	87.6%	11.7%	0.4%	0.3%
2016Q3	86.8%	12.5%	0.3%	0.4%
2016Q4	81.4%	18.2%	0.2%	0.2%
2017Q1	85.0%	14.7%	0.1%	0.1%

**Kuvio 3.** Suosituimmat mobiililaitteiden käyttöjärjestelmät (IDC 2017).

Sivustolla kartta tulee olemaan yksi sen tärkeimmistä osista. Sen käyttäminen tulee olla tutunomaista, kahdella sormella lähentämistä ja loitontamista. Oman sijainnin paikantaminen toimii kuten omassa tututussa karttapalvelussa. Vaatimuksena on käyttäjän sijaintitietojen salliminen laavusivustolla. Kuitenkin, jos sijaintitietoja ei haluta käyttää, voidaan laavujen lisäys tehdä kartan kautta. Laavujen selaaminen voi tapahtua ilman sijaintitietoja.

### **3 VAATIMUSMÄÄRITTELYT**

Ohjelmistokehityksessä vaatimusmäärittelyt ovat jotain, mitä tietokoneohjelman on kyettävä tekemään järjestelmän käyttäjän hyväksi. Vaatimusmäärittelyissä mietitään vaatimuksia asiakkaan kannalta, jolloin kysymys voisi olla: ”Mitä asiakas tarvitsee?” (Leppäniemi, 2012). Vaatimusmäärittelyt ovat siis jokin toiminto, ominaisuus tai periaate, jotka järjestelmän tulee tuottaa tai pystyä tekemään niin, että siitä olisi hyötyä (Gulak & Guiney 2000, 4).

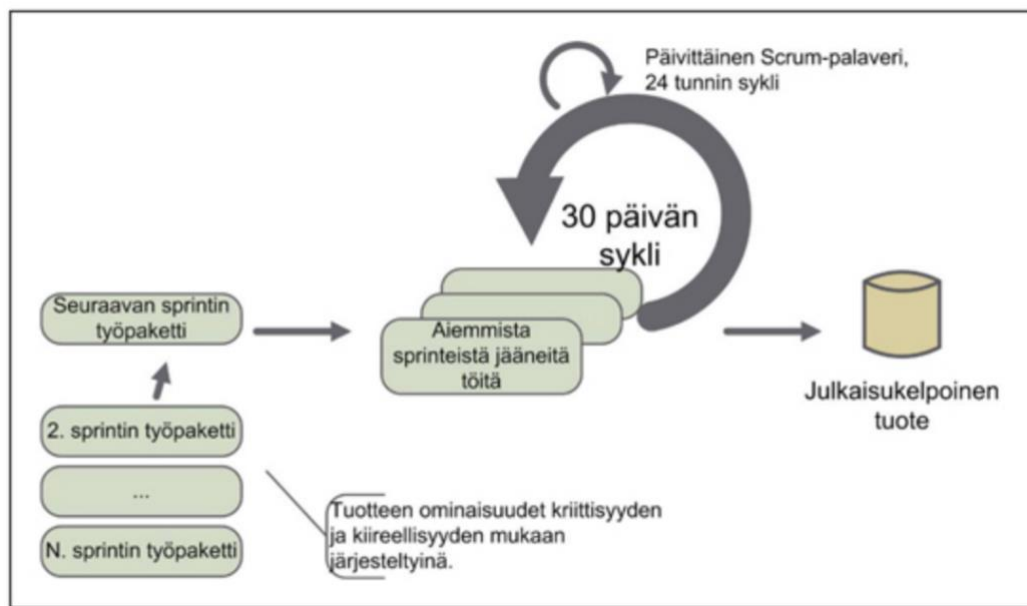
#### **3.1 Käyttäjäkertomukset**

##### **3.1.1 Scrum**

Käyttäjäkertomukset tai käyttäjätarinat ovat osa ketterää menetelmää eli scrumia. Scrum on suosittu työskentelymenetelmä ohjelmistokehityksessä. Tämä ketterä menetelmä auttaa tiimin jäseniä huomaamaan, mitä on tehty tai missä ollaan jäljessä ja mitä täytyy tehdä. Sen sijaan, että käytetään tunteja palaveriiniin, pidetään päivittäin pikainen katselmus jokaisen tiimin jäsenen työtilanteesta. Työt jaetaan lyhyen aikavälin päämääriin, jolloin oletuksena on, että projekti pysyy helpommin aikataulussaan ja tiedetään ajoissa, jos projekti onkin jäljessä.

Scrumin yksi sykli on noin kuukauden mittainen (Kuvio 4). Yhteen sykliin niputetaan tietty määrä ohjelmiston toimintoja tai ominaisuuksia ja tämän syklin aikana kyetään testaamaan, että kaikki toimii (Kasurinen 2013, luku 2).





**Kuvio 4.** Scrum-malli (Kasurinen 2013).

Scrumissa keskitytään yhteen osa-alueeseen kerrallaan, jolloin tehdään ensin yksi asia valmiiksi, sitten vasta siirrytään seuraavaan. Näin vältetään moniajolta (*multitasking*). Esimerkkinä, miten yhden asian hoitaminen loppuun ennen toisen asian aloittamista on tehokkaampaa, voitaisiin näyttää seuraavassa ryhmätyössä. Ryhmänjohtaja kirjoittaa paperille muiden jäsenten tehtävien nimet heidän antamiensa tietojen perusteella. He kertovat tehtäviensä nimet kirjain kirjaimelta vuorotellen. Otetaan aikaa, miten kauan menee, että jokaisen sana on kirjoitettu kokonaan paperille. Seuraavaksi verrataan sitä aikaa siihen, jos jokainen kertoo oman tehtävänsä nimekkeen kokonaisuudessaan. Ryhmänjohtaja kirjaa kokonaiset sanat paperille ja huomataan, että tämä tapa vie huomattavasti vähemmän aikaa kuin edellinen tapa.

### 3.1.2 Käyttäjäkertomus – user story

Aina ei vaatimusmäärittelyä tehdä täydellisesti tai kokonaan, vaan vaatimukset saattavat selvitä vasta projektin jo alettua. Yksi käytännöllinen tapa päästä vaatimusmäärittelyihin kiinni on tehdä käyttäjäkertomuksia. Kertomuksissa kerrotaan lyhyesti *mikä*, *miksi* ja *miten* niin, että myös asiakas ymmärtää ne.



Käyttäjäkertomuksia kirjataan ylös mahdollisimman monta erilaista kertomusta käyttäjien ja asiakkaiden kannalta. Kaikkia kertomuksia ei välttämättä toteuteta, mutta niistä voidaan saada myöhemmin tarkennuksia uusiin vaatimuksiin (Tolvanen & Puimala 2013).

Käyttäjäkertomusten apusanoina voidaan käyttää englanninkielisiä sanoja:

- AS A
- I WANT
- SO THAT

Nämä lyhyet lauseet kertovat juuri sen, kuka haluaa tehdä ja mitä. Joissain tapauksissa niihin voidaan kirjata lisätietoina hyväksymiskriteeri tai kommentteina esimerkiksi tavan, miten tämä onnistuisi.

Kun tein laavusivustolle käyttäjäkertomuksia, käytin apunani mahdollista tulevaa käyttäjää sekä jo olemassa olevia sovelluksia, kuten muun muassa Geocaching-sovellusta, jonka periaatteita ja tapoja pystyin soveltamaan myös laavusovellukseen (Taulukko 1).

**Taulukko 1.** Laavusivustoon tekemäni käyttäjäkertomukset ja -tarinat.

AS A... (toimijana)	I WANT... (haluan)	SO THAT... (että voin)
	Ulkoilijana haluan löytää laavun kartalta tietyltä alueelta, että löytäisin uusia laavupaikkoja.	
	Käyttäjänä haluan kirjautua sovellukseen, että voin lisätä uuden laavun sijainnin järjestelmään.	
	Käyttäjänä haluan kirjata uuden laavun sivustolle oman sijaintini mukaan, että muut löytävät laavun myös.	
	Käyttäjänä haluan kirjata uuden laavun sivustolle antamieni koordinaattien avulla, että muut löytävät laavun myös.	
	Käyttäjänä haluan kirjata uuden laavun sivustolle klikkaamalla karttaa, että muut löytävät laavun myös.	
	Käyttäjänä haluan ladata kuvia laavun profiiliin, että muut näkevät ne.	

Käyttäjänä haluan muokata laavun tietoja myöhemmin, että saisin laavun tiedot päivitettyä.
Käyttäjänä haluan hakea laavuja eri hakumenetelmin, esim. sijainnin mukaan tai vihjesanoilla, että löydän etsimäni laavun.
Käyttäjänä haluan tietää, miten tietylle laavulle pääsee, että tiedän, miten pitkä matka sinne on.
Käyttäjänä haluan tietää, miten tietylle laavulle pääsee, että tiedän, minne voin pysäköidä auton.
Käyttäjänä haluan tietää, millainen reitti laavulle on, että tiedän varautua.
Käyttäjänä haluan tietää, mitä varustuksia laavulla on, että tiedän varustautua tarpeen mukaan.
Käyttäjänä haluan hakea laavuja kartalta ilman kirjautumista järjestelmään, että voin nopeammin löytää laavun metsässä ollessani.
Käyttäjänä haluan saada tiedon, kun laittamani laavu on hyväksytty järjestelmään, että tiedän sanoa ystäväilleni.
Ylläpitäjänä haluan pystyä päivittämään ja hallinnoimaan laavutietoja sekä profiileja, että virheelliset tai epäasialliset päivitykset voidaan poistaa.
Ylläpitäjänä haluan saada viestin, kun uusi päivitys (uusi laavu, uusi kuva tai muutokset laavun tiedoissa) tulee sivustolle.
Käyttäjänä haluan poistaa oman profiilini, että se ei enää olisi järjestelmässä.
Käyttäjänä haluan muokata profiilini tietoja, että saan päivitettyä omat tietoni.
Käyttäjänä haluan tietää tietyn laavun tiedot laittaneen profilitiedot, että tiedän, mitä muita tietoja hän on lisännyt järjestelmään.
Käyttäjänä haluan kartan näkymän myös satelliittikuvana ja maastokarttana, että osaan paremmin paikallistaa sijaintini.
Käyttäjänä haluan merkitä omiin tietoihini laavut, joissa olen käynyt, että voin myöhemmin katsoa ne.
Käyttäjänä haluan vaihtaa asetuksia niin, etteivät muut näe minusta liikaa tietoja, koska haluan pitää yksityisyyteni.
Käyttäjänä haluan lisätietoa järjestelmästä, että voin lukea käyttöohjeet, käyttöehdot tai yksityisyydensuojasta.
Käyttäjänä haluan kirjoittaa kommentteja vierailemani laavun sivuille, että voin kiittää/valittaa/antaa muille tietoa siitä.

### 3.2 UML-notaatiostandardi

Käyttötapaukset ovat osa UML-mallinnuskieltä. Se tulee sanoista Unified Modeling Language. UML on graafinen merkitätapa, jolla dokumentoidaan järjestelmän vaatimuksia (Kulak & Guiney 2000, 26-27). Sen avulla voidaan esittää järjestelmän kuvaus siten, että myös henkilöt, sekä asiakkaat että käyttäjät, jotka eivät ole ohjelmoinnissa mukana, ymmärtävät sitä (Kasurinen 2013, luku 3). UML koostuu erilaisista kaaviotyypeistä, joita ovat esimerkiksi:

- Use case diagram eli käyttötapauskaavio
- Activity diagram eli aktiviteettikaavio
- Class diagram eli luokkakaavio
- Object diagram eli oliokaavio

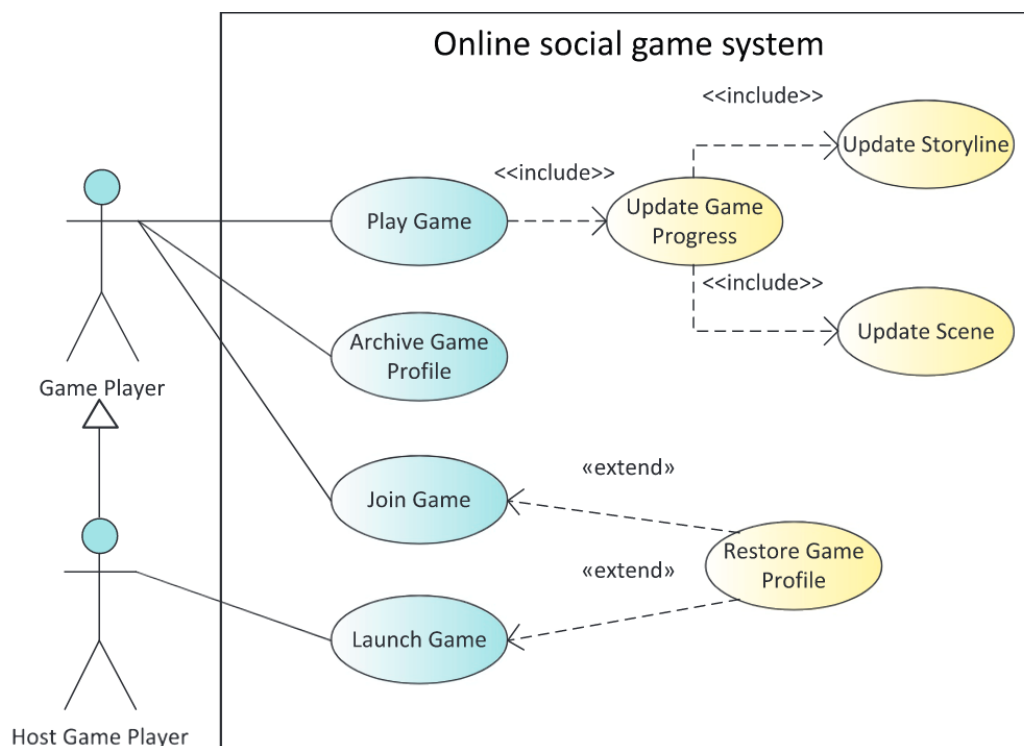
### 3.3 Käyttötapaus

Käyttötapaus käy läpi erilaisia tapahtumaketjuja, joita sovelluksen tai järjestelmän toiminnassa tapahtuu käyttäjien toimesta. Käyttötapaus on kaksiosainen, siinä on käyttötapauskaavio ja itse käyttötapaukset. Näiden läpikäynti auttaa sovelluksen suunnittelussa. Vaikka käyttötapaukset käydään läpi melko yksityiskohtaisesti, eivät ne sisällä silti toteutukseen liittyviä asioita.

#### 3.3.1 Käyttötapauskaaviot - use case diagram

Kaavioista käyttötapauskaaviot voidaan usein tehdä ensimmäisenä. Käyttötapaukset ohjaavat ja määrittelevät myös muita kaavioita, ja sitä, miten koko systeemi toimii vuorovaikutuksessa kaikkien muihin ympärillä, kuten käyttäjiin sekä muihin järjestelmiin ja tekijöihin (Kulak & Guiney 2000, 28-29). Käyttötapauskaavioissa kuvataan eri käyttäjärooleja eri tapahtumissa tai käyttötapauksissa.

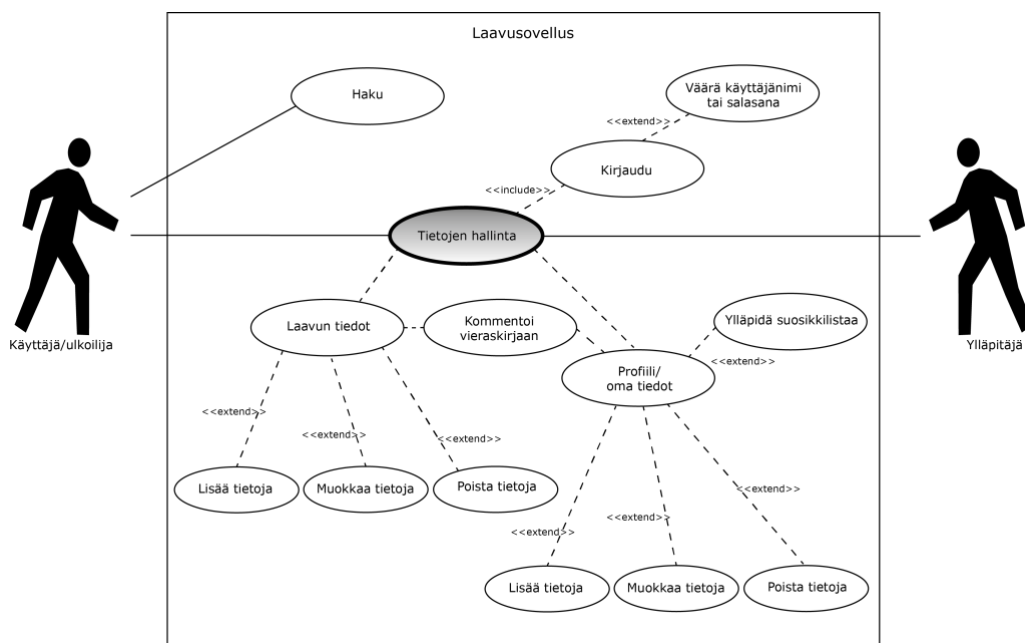
Kaavioiden käyttäjät kuvataan useimmiten yksinkertaisina tikku-ukkoina. Käyttäjät yhdistetään viivalla käyttötapauksiin, jotka ovat kuvattuna ellipsin muotoisina. Järjestelmä itse on neliön sisällä.



**Kuvio 5.** Esimerkki käyttötapauskaaviosta (Xiaocong 2015, 204).

Esimerkkikaaviossa kuviossa 5 on kuvattu verkkopelin järjestelmä laatikkona, jonka ulkopuolelle on piirretty kaksi erilaista käyttäjää, niin sanonut *actorit*, pelaaja sekä *host*. Käyttäjistä lähtevät viivat pääkäyttötapauksiin. Katkoviivalla merkityt viivat ovat niin sanottuja alikäyttötapauksia, jotka sisältyvät pääkäyttötapaukseen. Nämä merkitään merkinnällä `<<include>>`. Käyttäjän pelaaman pelin tiedot päivittyvät järjestelmään. Kaaviossa voidaan käyttää myös merkintää `<<extend>>`, joka liittyy toiseen käyttötapaukseen. Se on niin sanottu laajennus pääkäyttötapahtumasta.

Itse laavusovelluksen käyttötapauskaavio on esitetty kuviossa 6. Käyttäjinä ovat ulkoilija sekä ylläpitäjä. Kaaviosta voidaan todeta, että ulkoilija pääsee hakemaan tietoa ilman kirjautumista sekä hallitsemaan tietoja kirjautumalla sivustoon. Tietojen hallintaan liittyy sekä laavun että käyttäjän omien tietojen lisäys, muokkaus ja niiden poistaminen. Laajennuksena muun muassa kirjautumiseen on liitetty merkintä `<<extend>>` väärä käyttäjänimi tai salasana. Ylläpitäjä hallitsee kaikkia tietoja järjestelmässä.



**Kuvio 6.** Laavusovelluksen käyttötapauskaavio.

### 3.3.2 Käyttäjien analysointi ja käyttötapaukset – use cases

Käyttäjiä laavusovelluksessa ovat kirjautunut jäsen, ylläpitäjä ja kirjautumaton jäsen. Kirjautunut jäsen on henkilö, joka on kirjautunut järjestelmään. Kirjautuneena hän pystyy muokkaamaan omia tietojaan, lisäämään uusia laavuja, muokkaamaan lisäämiensä laavujen tietoja, lisäämään niihin valokuvia, poistamaan lisäämänsä laavun sekä poistamaan oman profiilinsa sekä kirjoittamaan kommenttinsa laavun sivuille tai vieraskirjaan.

Ylläpitäjä on henkilö tai taho, joka pitää laavusovelluksen tietoja ajan tasalla, hän hyväksyy uudet lisätyt laavut ja päivitykset ja toimii eräänlaisena sivuston valvojana.

Kirjautumaton jäsen on henkilö, joka käyttää selaimella tai sovelluksella laavun järjestelmää. Hän pystyy selaamaan laavuja kartalla sekä näkee laavuihin annetut tiedot. Jos hän on antanut luvan käyttää sijaintitietojaan, pystyy hän näkemään myös oman sijaintinsa karttanäkymässä.

Käyttötapauksissa ei käydä läpi kaikkia tapauksia, vaan muutama esimerkkitapaus, joihin on upotettu useampi eri toiminto. Seuraavassa esitellään laavusivuston eri käyttötapauksia:

### **Käyttötapaus 1**

Nimi:	Laavun lisääminen järjestelmään.
Suorittaja:	Käyttäjä/retkeilijä.
Esiehdot:	Käyttäjä on kirjautunut järjestelmään.
Kuvaus:	Käyttäjä on retkeillyt laavulle, jonka sijaintia ei ole merkitty järjestelmään. Käyttäjä hakee kartan avulla oman sijaintinsa ja lisää uuden laavun järjestelmään. Koordinaatit sekä lisäyspäivämäärä tulevat automaattisesti tietoihin. Laavun nimi sekä lisätiedot laavusta käyttäjä kirjaa itse. Lisätietoina hän kirjaa muun muassa parkkipaikan tiedot ja reitin pituuden sieltä sekä polttopuiden saatavuuden. Käyttäjä ottaa laavusta myös valokuvan, jonka lisää myös laavun tietoihin. [Poikkeus: laavun lisääminen ei onnistu].
Poikkeukset:	Laavun lisääminen ei onnistu: Järjestelmävalvojan tulee hyväksyä uudet lisäykset ennen kuin se näkyy järjestelmässä. Tästä tulee käyttäjän näytölle tieto.
Lopputulos:	Uusi laavu on lisätty tietoineen karttanäkymälle hyväksymisen jälkeen.
Muut vaatimukset:	Järjestelmänvalvojan hyväksyminen tulee tapahtua 24 tunnin sisällä laavun lisäämisestä.

### **Käyttötapaus 2**

Nimi:	Laavun tietojen muokkaaminen.
Suorittaja:	Käyttäjä/retkeilijä.
Esiehdot:	Käyttäjä on kirjautunut järjestelmään.
Kuvaus:	Käyttäjä menee omaan profiiliinsa ja hakee listan laavuista, jotka hän on itse lisännyt sivustolle. Avaa tietyn

laavun tiedot ja muokkaa laavun sijaintia ja lisää valokuvan sekä muuta tietoa.

Lopputulos: Muutokset näkyvät heti järjestelmässä. Järjestelmänvalvoja saa muutoksista tiedon, käy tarkastamassa tiedot ja jättää muutokset voimaan.

### Käyttötapaus 3

Nimi: Käyttäjä unohtaa salasanan.

Suorittaja: Käyttäjä/retkeilijä.

Esiehdot: Käyttäjällä on sivusto auki selaimessa mobiililaitteessaan.

Kuvaus: Käyttäjä yrittää kirjautua järjestelmään tunnuksillaan, mutta joko tunnus tai salasana on väärin. Painaa linkkiä: ”Unohditko salasanasasi?”, jonka näkymässä on tekstilaatikat tunnukselle ja sähköpostille. Käyttäjä muistaa tunnuksensa, joten kirjoittaa sen ja painaa lähetä, minkä jälkeen tulee ilmoitus, jossa uuden salasanan vaihtamiseksi avattava linkki lähetetään sähköpostiosoitteeseen (esim. l...a@gmail.com). Käyttäjä avaa sähköpostiin tulleen linkin. Järjestelmä pyytää vaihtamaan uuden salasanan, käyttäjä varmistaa sen ja pystyy nyt kirjautumaan sivustolle uudelleen. [Poikkeus: salasanalinkin lähettäminen ei onnistu]

Poikkeukset: Uuden linkin lähettäminen ei onnistu, sillä tunnus oli väärä. Käyttäjä saa tästä tiedon, joten voi laittaa oikean tunnuksen tai sähköpostiosoitteen.

Lopputulos: Käyttäjä pystyy kirjautumaan järjestelmään salasanan vaihtamisen jälkeen.

Muut vaatimukset: Sähköpostiin lähetettävän viestin tulee lähteä välittömästi käyttäjälle.

### Käyttötapaus 4

Nimi: Käyttäjä selaa laavuja kartalla.

Suorittaja:	Ei kirjautunut käyttäjä/retkeilijä.
Esiehdot:	Käyttäjällä on laavusivusto auki mobiililaitteessaan.
Kuvaus:	Käyttäjä tutkii sivustolla olevaa karttaa ja siihen lisättyjä laavuja. Hän lähentää ja loitontaa Suomen karttaa ja aktivoi tietyn laavun, josta näkee kaikki annetut tiedot, kuten valokuvat, koordinaatit, lisätiedot laavusta, lisääjän tunnuksen ja lisäyspäivämäärän.
Poikkeukset:	Kartta ei löydä käyttäjän omaa sijaintia. Käyttäjän täytyy sallia sijainnin käytön tälle sivustolle.
Lopputulos:	Käyttäjät pystyvät selaamaan ja hakemaan laavujen tietoa sivustolla.

### Käyttötapaus 5

Nimi:	Käyttäjän rekisteröityminen.
Suorittaja:	Uusi käyttäjä/retkeilijä.
Esiehdot:	Käyttäjällä on laavusivusto auki.
Kuvaus:	Käyttäjä valitsee kirjautumiskuvakkeen, jossa rekisteröitymislinkki järjestelmään. Antaa sähköpostiosoitteensa ja tunnuksen tai nimen.
Poikkeukset:	Annettu tunnus on väärä, sähköpostiosoite on jo käytössä tai salasana vaatimukset eivät täyty. Näistä kaikista tulee ilmoitus, jolloin käyttäjä voi kirjata sopivat tiedot.
Lopputulos:	Käyttäjä on rekisteröitynyt järjestelmään ja pystyy lisäämään uusia laavuja.

### Käyttötapaus 6

Nimi:	Suosikkilistan ylläpitäminen.
Suorittaja:	Käyttäjä/retkeilijä.
Esiehdot:	Käyttäjä on kirjautunut järjestelmään ja aktivoinut laavun kartalta.
Kuvaus:	Käyttäjä valitsee laavun tiedoista ”lisää suosikkeihin”. Laavu tallentuu käyttäjän suosikkilistalle. Laavun



tiedoissa näkyy kuvake, joka kertoo, onko laavu lisätty suosikkilistalle vai ei.

Poikkeukset: Laavu on jo lisätty suosikkilistalle. Jos käyttäjä yrittää lisätä uudelleen saman laavun, tulee kysymys, haluaako käyttäjä varmasti poistaa kyseisen laavun suosikkilistalta.

Lopputulokset: Valittu laavu on lisätty käyttäjän suosikkilistalle.

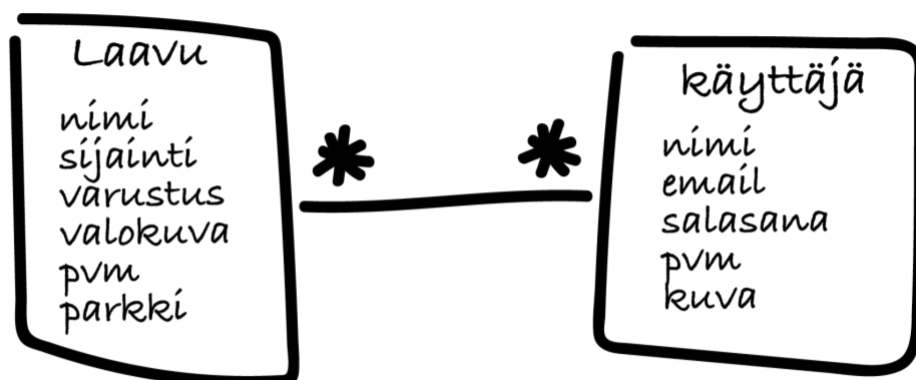
## 4 TIETOKANTA

Myös tietokantojen suunnittelu on aikaa vievää, mutta erittäin tärkeää. Tietokannat kattavat kaikki tiedot, joita sovelluksesta talletetaan. Tietokannassa tietoina ovat esimerkiksi käyttäjän ja laavun kaikki tiedot. Aina käyttäjän nimestä, salasanaa ja profiilikuvasta laavun varusteluihin ja sijaintiin saakka.

Tietokanta siis on joukko tietoja, jotka loogisesti kuuluvat toisiinsa. Tietokantoja voidaan käsitellä tietokantakielillä, joista yksi tunnetuimmista on SQL. Se, millä näitä tietoja käsitellään, kutsutaan tietokannanhallintajärjestelmäksi. Näitä ovat esimerkiksi Oracle, Microsoft SQL Server, MySQL ja Access. Ilman tietokantoja jouduttaisiin käyttämään tiedostoja, joita tulisi olemaan liikaa ja eheys saattaisi olla heikompaa. Eheydellä tarkoitetaan sitä, kun kaikki tiedot tietokannassa ovat keskenään yhteensopivia sekä oikeita. Tietokannan ansiosta useat ihmiset ja sovellukset voivat käyttää näitä samoja tietoja yhtä aikaa. Ne myös päivittyvät reaaliaikaisesti (Hovi, Huotari & Lahdenmäki 2003, 4-5, 11).

### 4.1 Taulut

Aloitin tietokannan suunnittelun tauluista, jotka ovatkin tietokannan peruste. Perustauluina olivat laavu- ja käyttäjätaulut (Kuvio 7). Käyttäjäkertomuksia hyväksikäyttäen mietin, mitä tietoja tarvitaan tietokannassa. Myös listanäkymän hahmotteleminen auttaa suunnittelussa. Hankalinta oli miettiä, miten tiedot sijoitetaan tauluihin. Taulujen määrä kasvaa, kun mietin, miten esimerkiksi koordinaattien tai laavun varustusten tiedot saadaan ja miten ne talletetaan.



**Kuvio 7.** Laavusivuston hahmotelma, jossa kaksi päätaulua.

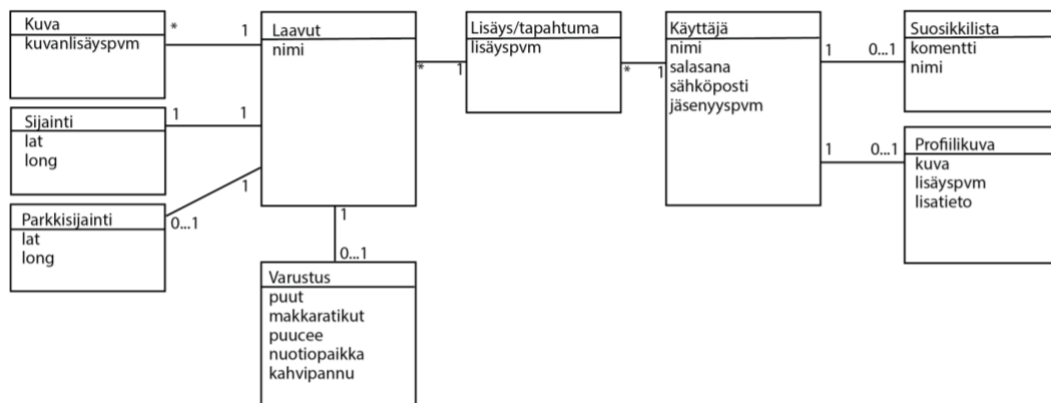
Kuviossa 7 hahmottelin ensin päätaulut. Tähtimerkintä (\*) tarkoittaa, sitä että yhdellä käyttäjällä voi olla monta laavua sekä laavulla monta käyttäjää.

Taulukossa 2 on esimerkki listanäkymästä, jossa on eri tietoja eri tauluista. Lista on siten mahdollisuus saada kaikki nämä tiedot eri tauluista. Taulujen välille tulee laittaa kytkös, että taulut osaavat lukea toisiaan. Näitä kytköksiä kutsutaan avaimiksi. PK ja FK eli *primary key* sekä *foreign key*. Avaimien tietoja ei tarvitse näyttää käyttäjille, mutta ne ovat pakollisia yhteyksien lukemiseksi.

**Taulukko 2.** Listanäkymä laavun tiedoista.

nimi	lisäyspvm	sijainti	parkki	lisännyt	lisätieto	varustus	vieraskirja	kuva
Metsä laavu	27.11.2017	[koordinaatit]	[koordinaatit]	Ulkoilija3	Hyvä peruslaavu, ei kovin selkeä reitti perille	puut, nuotiopaikka, makkaratikut	Paistoin makkarat	[kuva]

Luokkakaaviokuviossa mietin jo hieman pidemmälle tauluja ja niiden tietoja (Kuvio 8). Kun katsotaan tarkemmin eri tauluja, kuten esimerkiksi laavu- sekä sijaintitaulua, huomataan, että yhden laavun tiedoissa on pakollisena tietona sijaintitiedot. Sen sijaan sijaintitietoja ei voi olla olemassa ilman laavua. Otetaan toiset taulut, esimerkiksi laavu- sekä parkkisijaintitaulut. Laavulla saa olla parkkitiedot, mutta eivät ole pakollisia. Toisin päin katsottuna, jos parkkisijainnilla on tietoja, on oltava myös laavun tiedot.



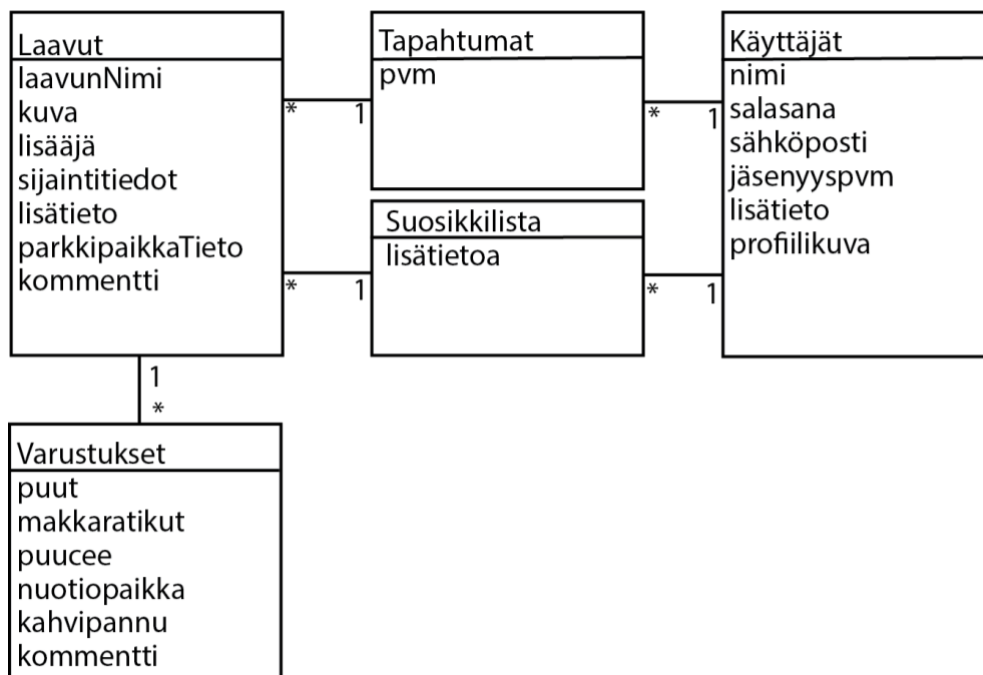
**Kuvio 8.** Laavusivuston tietokannan luokkakaavio.

Yhteyksien pakollisuus voidaan merkitä usealla tavalla. Itse käytin UML-tapaa, jossa merkintöinä ovat numerot ja tähti. Taulukossa 3 on selitykset näille merkinnöille.

**Taulukko 3.** Luokkakaavion merkintöjen selitykset.

Merkintä	Selitys
0...1	Taulun tietoja ei välttämättä ole pakko olla, mutta enimmillään saa olla vain yksi tieto.
1	Tämän taulun tietoja on pakko olla tasan yksi.
* (sama kuin 0...*)	Tietoja voi olla monta tai ei yhtään.

Tietokanta voidaan tehdä myös yksinkertaisemmalla tavalla. Jos taulujen suhteet toisiinsa ovat yhden suhde yhteen, voidaan tiedot tallentaa ainoastaan yhteen tauluun. Kuviossa 9 tauluja on yhdistetty toisiinsa, mutta tiedot pysyvät samoina.



**Kuvio 9.** Muokattu luokkakaavio laavusivuston tietokannasta.

## 4.2 Tietotyypit















Kun sain luokkakaavion hahmoteltua, saatoinkin miettiä tietokantaa jälleen tarkemmin. Nimesin jokaisen taulun yksittäiset tiedot ja pohdin, missä muodossa ne talletetaan. Suunnittelin siis ovatko ne pelkkiä numeroita tai päivämääriä vai sekalaisia merkkejä. Eli mitä tietotyyppejä ne tulevat olemaan (Taulukko 4). Jokainen taulu tulee saamaan myös yksilöidyn ID-merkinnän, joka tulee olemaan taulun avain.

**Taulukko 4.** Tietotyyppejä selityksineen.

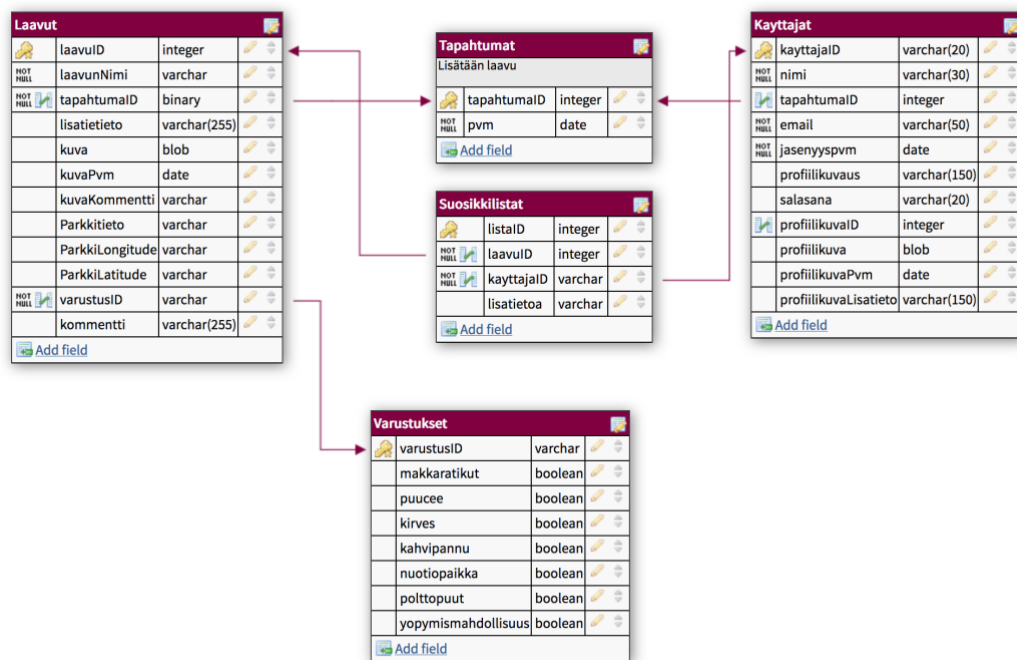
Tietotyyppi	Selitys
varchar	Merkkitieto, joka voi maksimissaan olla 255 merkkiä pitkä. Merkitään esim. varchar(30)
integer/int	Kokonaisluku, jonka maksimipituus voidaan määritellä: int(11). Minimipituus: -2147483648, maksimipituus: 2147483647.
date	Vuosi, kuukausi ja päivä. YYYY-MM-DD.
boolean	Tosi tai epätosi.

tinyint	Lyhyt numerotieto, voidaan käyttää tosi/epätosi -arvoina. 0=epätosi, 1=tosi.
blob	Binääridata, jota voidaan käyttää mm. valokuvan tallentamiseen.

Käytin tietokannan graafisessa suunnittelussa apuna DbDesigneria, joka on selaimessa käytettävä ilmaisohjelma. Kaavion tekeminen tällä ohjelmalla oli helppoa ja samalla se auttoi kokonaisuuden miettimisessä. Kaavion tauluihin pystyi merkitsemään nämä tietotyypit sekä avaimet. Kaikkia tietoja ei ole pakko lisätä tietokantaan, mutta joidenkin tietojen pakollisuus on määrätty etukäteen NOT NULL -merkinnällä, jolloin kyseinen tieto ei voi jäädä tyhjäksi. Avaimien määrittelyssä voidaan käyttää automaattilaskurina AUTO\_INCREMENT (Kuviot 10-11).

Laavut			
	laavuID	integer	
NOT NULL	laavunNimi	varchar	
NOT NULL	tapahtumaID	binary	
	lisatietieto	varchar(255)	
	kuva	blob	
	kuvaPvm	date	
	kuvaKommentti	varchar	
	Parkkitieto	varchar	
	ParkkiLongitude	varchar	
	ParkkiLatitude	varchar	
NOT NULL	varustusID	varchar	
	kommentti	varchar(255)	
 Add field			

**Kuvio 10.** Laavutaulun tiedot sekä pääavain.



**Kuvio 11.** Tietokannan taulukaavio DdDesignerissa.

## 5 VISUAALINEN SUUNNITTELU

Nykyään sivustot täytyy suunnitella responsiivisiksi. Responsiivisella tarkoitetaan sitä, että verkkosivut skaalautuvat näytön mukaan käyttäjälle miellyttävämmäksi. Eli sivujen tulee toimia sujuvasti sekä mobiililaitteissa että kotikoneella.

Verkkosivujen navigointi onkin yksi tärkeimpiä asioita käyttäjälle. Pienemmälle ruudulle ei saa mahtumaan niin montaa linkkiä tai painiketta kuin isommalle, siksi responsiivisten verkkosivujen suunnittelu kannattaakin aloittaa pienempiä ruutuja ajatellen ensin (Myers 2015. Kappale 3).

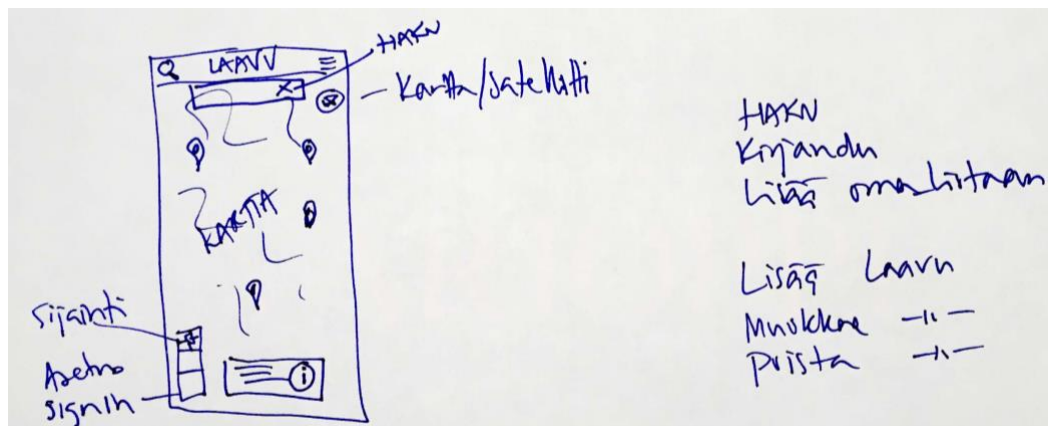
Responsiivisuus on siis se, mitä täytyy miettiä, kun suunnitellaan mallipohjaa sivuille. Onneksi tähän löytyy paljon apuja eri sivustoilta. Varmasti yksi suosituimmista responsiivisista työkaluista, joilla on valmis kirjasto koodeineen, on Bootstrap. Tätä voisi käyttää hyväksi myös laavusivustojen suunnittelussa. Toisaalta, koodissa tulee mukana paljon ylimääräistä, mitä ei pakosti tarvita. Siksi voidaan miettiä myös muita kirjastoja, joita löytyy eri verkkosivuilta tai jopa oman kirjaston kehittämistä. Tähänkin löytyy tukea muun muassa Chelsea Myersin kirjasta *Responsive Web Design Patterns*. Hän antaa koodinsa vapaasti käytettäväksi alkuun pääsemiseksi oman kirjaston luomisessa (Myers 2015. Kappale 1).

### 5.1 Wireframe – rautalankamalli

Wireframe eli rautalankamalli on karkea suunnitelma sovelluksen näkymästä ruudulla. Siinä ei tarvitse vielä miettiä graafista suunnitelmaa, vaan lähinnä, miten painikkeet sekä yleiseen käytettävyyteen vaikuttavat asiat ja sisältö sijoittuvat (Thornsby 2016).

Rautalankamallin voi suunnitella joko paperille tai myös käyttää apunaan sovelluksia ja sivustoja, joita on paljon tarjolla myös ilmaiseksi. Itse piirsin ensin paperille ja sitten vasta työstin tietokoneella malleja (Kuvio 12). Vertailin usean eri sivuston ja sovelluksen näkymiä keskenään ja omaksuin niistä joitain parhaimpia ideoita, joita sovelsin laavusivuille.



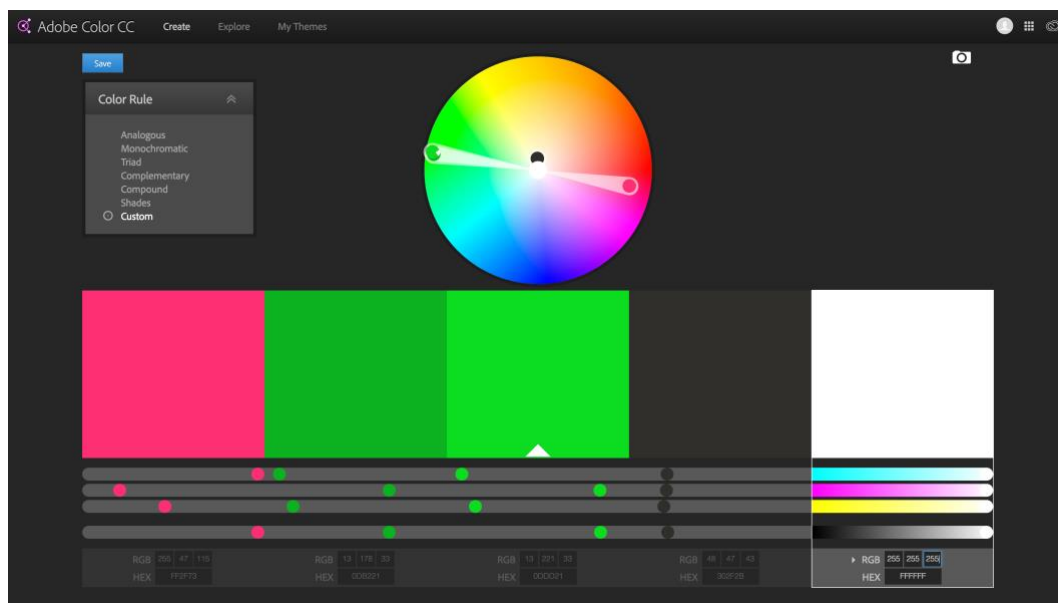


**Kuvio 12.** Karkea ensimmäinen rautalankamallisuunnitelma sivustosta.

## 5.2 Fontti ja värimaailma

Fontiksi valitsin Helvetica Lightin, joka on hyvin yleinen fontti, siten saatavilla helpommin ja niin sanottu *turvallinen* fontti verkkosivuilla. Lisäksi se on tyypillinen selkeä fontti, joka on tuttu monelle, siten sen lukeminen helppoa. Se ei tuota mielipahaa tai negatiivisia ajatuksia kenellekään.

Värien miettimisessä käytin apunani Adobe Color CC:tä (entinen Kuler) (Kuvio 13). Koska kartta on pääelementti sivuilla, määräytyy sivun värit myös kartan mukaan. Aloitussivulla väreinä näemme siis lähinnä valkoista sekä tumman ruskeaa tekstin värinä. Väreinä halusin käyttää maanläheisiä värejä. Väripaletissa onkin kahta eri vihreää, niiden vastaväri punainen sekä valkoinen ja tumman ruskea fonttien ja taustan väreiksi.



**Kuvio 13.** Värien suunnittelua Adobe Color CC -ohjelmassa.

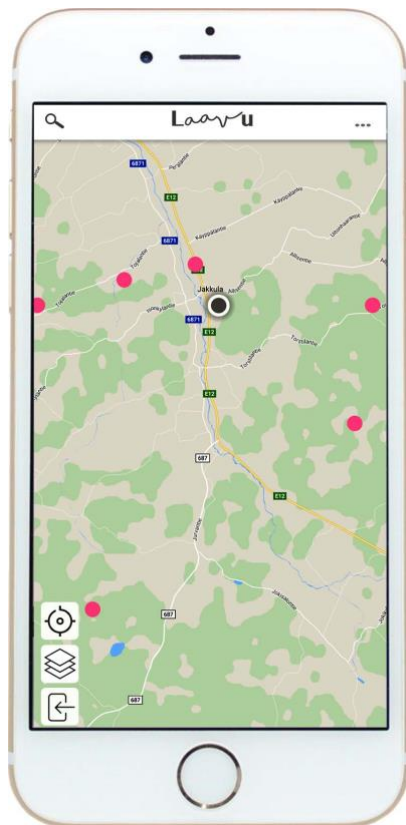
### 5.3 Visuaalinen suunnittelu

Käytin Photoshopia ja Illustratoria tarkemman näytösuunnitelman sekä logojen tekemiseen. Ensin mietin väripalettia ja sen jälkeen oletussivun näyttöä. Vertailin myös tässä tapauksessa muiden sivustojen ja sovellusten näkymiä keskenään ja käytin niitä antamaan inspiraatiota laavusovelluksen näyttöihin.

Suunnittelin myös kaksi logoa, jotka eivät välttämättä ole viimeiset versiot, sillä mahdollisen asiakkaan mielipidettä kuulisin mielelläni myös (Kuvio 14). Myös sivuilla käytössä olevat ikonit saattavat muuttua. Niitäkin on tarjolla useampia erilaisia. Myös omien ikonien tekeminen on mahdollista.



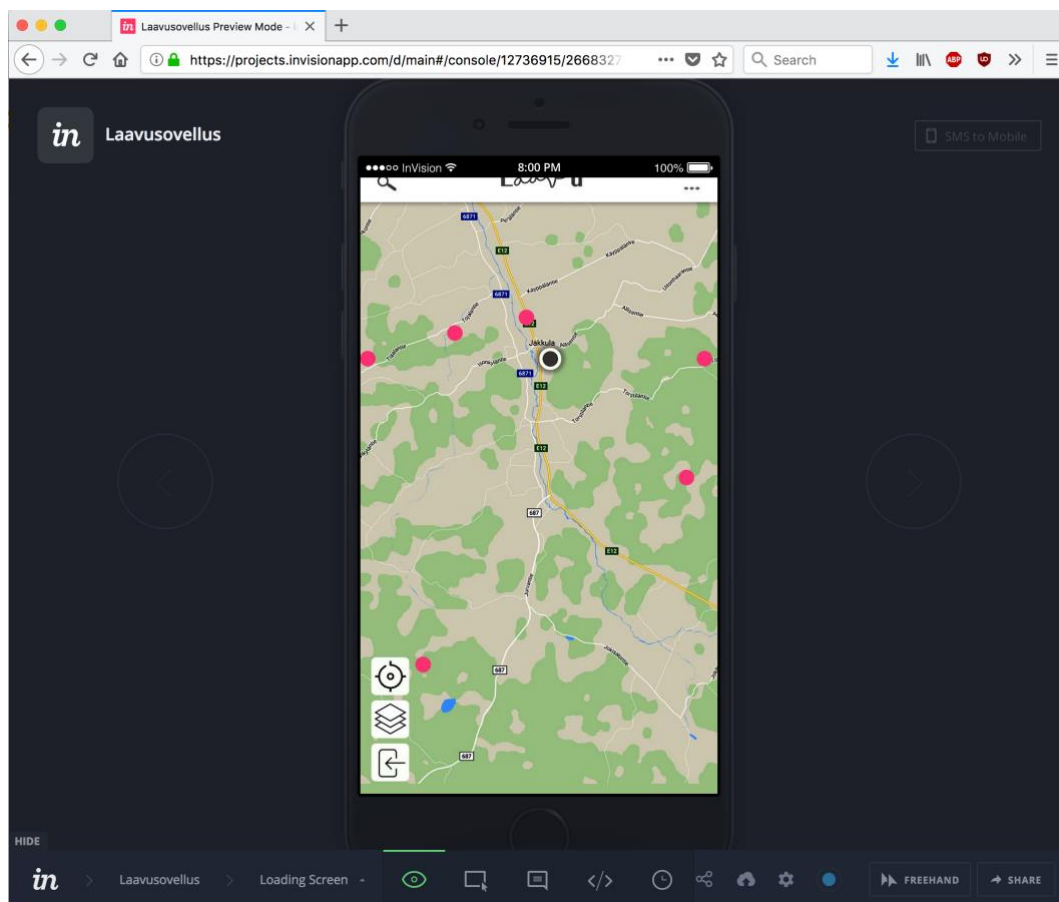
**Kuvio 14.** Laavusovelluksen kaksi eri logo-vaihtoehtoa.



**Kuvio 15.** Laavusovelluksen oletusnäkökulma puhelimessa.

### 5.3.1 Invision

Kun asiakkaalle näytetään sovelluksen suunnitelmaa, voidaan käyttää selainpohjaista Invisionia. Sen avulla on helppo näyttää kaikki toiminnot, joita sivustolle on suunniteltu. Sinne ladataan eri sivujen toteutetut näyttökuvat. Kuviin voidaan tehdä niin sanotut *hotspotit*, jolloin painikkeet toimivat ikään kuin oikealla sivustolla. Invision on interaktiivinen prototyyppi, jonka avulla asiakas saa oikean kuvan käytännössä, miltä sovellus näyttää ja miten sivut toimivat (Kuvio 16).



**Kuvio 16.** Invisionin näkymä.

## 6 POHDINTA

Tavoitteena tässä opinnäytetyössä oli tehdä suunnitelma laavusovellukselle tai -sivustolle jälkikehittämistä varten. Ennen kuin pystyy aloittamaan itse ohjelmoinnin, täytyy olla tiedossa tarkalleen, mitä tehdään. Työssä on käyty läpi käyttäjäkertomuksista aina tietokantaan ja rautalankamalleihin saakka.

Haastavinta tässä työssä oli luokkakaavion sekä tietokannan suunnitteleminen. Näiden parissa kului yllättävän paljon aikaa, eikä silloinkaan tiennyt, oliko tehnyt nämä oikein. Kaavioista ja tietokannasta tuli monta eri versiota, joita pyörittelin ja muokkailin useaan otteeseen. Onneksi ohjaavalta opettajalta pystyi kysymään neuvoa. Tietokantoihin liittyviä materiaaleja oli saatavilla sekä kirjastossa että verkossa. Se, että osaa soveltaa näistä esimerkeistä omaan järjestelmään, on varsin hankalaa.

Kun suunnittelee sovellusta, jonka itse haluaa myös toteuttaa, tulee suunnitteluvaiheessa miettineeksi liikaa ohjelmointivaihetta. Jos suunnitelmassa on nyt näin, miten sen sitten osaa ohjelmoida? Tämä oli ajattelutapa, jota ei olisi saanut käyttää. Toteutusta ei saa miettiä suunnitteluvaiheessa!

Tavoitteena – tai toiveena – on, että tämä opinnäytetyö tulee jossain vaiheessa vielä käyttöön. Sovelluksen ohjelmointi tulee olemaan projektina suhteellisen iso, joten on hyvä, että suunnitelmat ovat jo valmiina.

## LÄHTEET

Haikala, I. & Märijärvi, J. 2002. Ohjelmistotuotanto. Helsinki. Talentum Media Oy.

Kulak, D. & Guiney, E. 2000. Use Cases: Requirements in Context. New York. ACM Press.

Kasurinen, J.P. 2013. Ohjelmistotestauksen käsikirja. Jyväskylä. Docendo.

Leppäniemi, P. 2012. Käytännön ohjelmistotuotanto. Viitattu 16.11.2017.  
<http://ohjelmistotuotanto.panuleppaniemi.com/>

Tolvanen, P. & Poimala, S. 2013. Ketteryys haltuun: Yleisimmät ketterät käytännöt. Sininen Meteoriitti. Viitattu 16.11.2017. <https://www.meteoriitti.com/2013/06/06/ketteryys-haltuun-yleisimmat-ketterat-kaytannot/>

International Data Corporation (IDC). 2017. Smartphone OS Market Share, 2017 Q1. Viitattu 21.11.2017. <https://www.idc.com/promo/smartphone-market-share/os>

Hovi, A., Huotari, J. & Lahdenmäki, T. 2003. Tietokantojen suunnittelu & indeksointi. Jyväskylä. Docendo.

Thornsby, J. 2016. Android UI Design. Birmingham. Packt Publishing.

Myers, C. 2015. Responsive Web Design Patterns. Birmingham. Packt Publishing.

Xiaocong, F. 2015. Real-Time Embedded Systems. Oxford. Newnes.